Improving Word Alignment Using Alignment of Deep Structures*

David Mareček

Institute of Formal and Applied Linguistics,
Charles University in Prague
marecek@ufal.mff.cuni.cz

Abstract. In this paper, we describe differences between a classical word alignment on the surface (word-layer alignment) and an alignment of deep syntactic sentence representations (tectogrammatical alignment). The deep structures we use are dependency trees containing content (autosemantic) words as their nodes. Most of other functional words, such as prepositions, articles, and auxiliary verbs are hidden. We introduce an algorithm which aligns such trees using perceptron-based scoring function. For evaluation purposes, a set of parallel sentences was manually aligned. We show that using statistical word alignment (GIZA++) can improve the tectogrammatical alignment. Surprisingly, we also show that the tectogrammatical alignment can be then used to significantly improve the original word alignment.

1 Introduction

Alignment of parallel texts is one of the well-established tasks in NLP (see [1] and [2]). It can be used for various purposes, such as for creating training data for machine translation (MT) algorithms, for extracting bilingual dictionaries, and for projections of linguistic features from one language to another.

In tectogrammatics (as introduced in Functional Generative Description by Sgall [3], and implemented in the Prague Dependency Treebank [4]), each sentence is represented by a tectogrammatical tree (t-tree for short). T-tree is a rooted dependency deep-syntactic tree. Example of an English t-tree is shown in Figure 2. Unlike in the surface syntax, only content (autosemantic) words have their own nodes in t-trees. Function words such as auxiliary verbs, subordinating conjunctions, articles, and prepositions are represented differently: for instance, there is no node representing auxiliary verbs *has* and *been* in the t-tree example, but one of the functions they convey is reflected by attribute *tense* attached to the autosemantic verb's node (*set*). Other attributes describe several cognitive, syntactic and morphological categories.

Our motivation for developing tectogrammatical alignment system is following. First, we need aligned dependency trees for experimenting with statistical dependency-based MT. Second, we want to collect evidence for the hypothesis that typologically

* The work on this project was supported by the grants GAUK 9994/2009, GAČR 201/09/H057, and GAAV ČR 1ET101120503.

V. Matoušek and P. Mautner (Eds.): TSD 2009, LNAI 5729, pp. 56–63, 2009.

different languages look more similar on the tectogrammatical layer, and thus the alignment of tectogrammatical trees should be "simpler" (in the sense of higher interannotator agreement and higher achievable performance of automatic aligners evaluated on manually aligned data). Third, we show that the alignment gained using the tectogrammatical analysis can be helpful for improving quality of the automatic alignment back on the word layer.

Several works already came with the idea of aligning content words only. Haruno and Yamazaki [5] argued that in structurally different languages (Japanese and English in their case) it is not feasible to align functional words, because their functions are often very different. Menezes and Richardson [2] use so called "logical forms" (LFs) – an unordered graphs representing the relations among the meaningful elements of sentences. These structures are very similar to tectogrammatical trees. Nodes are identified by lemmas of the content words and labeled arcs indicate the underlying semantic dependency relations between nodes. Watanabe, Kurohashi, and Aramaki [6] have also similar sentence structures and use an algorithm which finds word correspondences by consulting a bilingual dictionary.

The remainder of the text is structured as follows. In Section 2, we describe the manually annotated data. The evaluation metric we use is in Section 3. The GIZA++ alignment is described in Section 4. Our tectogrammatical aligner is presented in Section 5. The way how to improve GIZA++ word alignment using our tectogrammatical aligner is the subject of Section 6. Section 7 concludes and discusses future work.

2 Manual Word Alignment

For the purpose of training and evaluating alignment systems, we compiled a manually annotated data set. It consists of 2,500 pairs of sentences[1] from several different domains (newspaper articles, E-books, short stories and also from EU law). Quantitative properties of the annotated data sets are summarized in Table 1. PCEDT parallel corpus [7] contains a subset of English sentences from Penn Treebank corpus which were translated into Czech (sentence by sentence) for the MT purposes.

Each sentence pair was manually aligned on the word level independently by two annotators. The task was to make connections (links) between Czech and English corresponding tokens. Following Bojar and Prokopová [8] we used three types of connections: *sure* (individual words match), *phrasal* (whole phrases correspond, but not literally), and *possible* (used especially to connect words that do not have a real equivalent in the other language but syntactically clearly belong to a word nearby). An example is presented in Figure 1.

The "golden" alignment annotation was created from the two parallel annotations according to the following rules: a connection is marked as *sure* if at least one of the annotators marked it as *sure* and the other also supported the link by any connection

[1] Boundaries of Czech and English sentences did not always match. (i.e., there were not only 1:1 sentence relations). In such cases we either split the sentence in one language or join several sentences in the other language in order to have only 1:1 sentence relations in the annotated data.

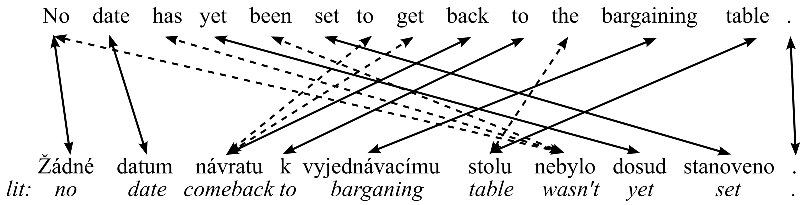
No date has yet been set to get back to the bargaining table .
Žádné datum návratu k vyjednávacímu stolu nebylo dosud stanoveno .
lit: no date comeback to barganing table wasn't yet set .

Fig. 1. Word-layer alignment (possible links are dashed)

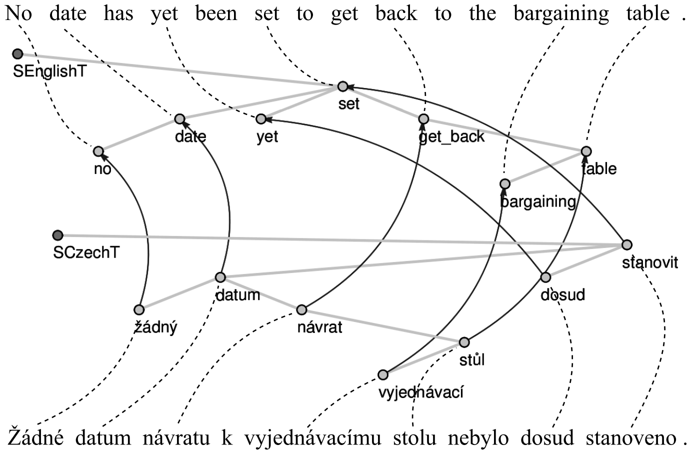
No date has yet been set to get back to the bargaining table .
SEnglishT
set
date
yet
get_back
table
no
bargaining
SCzechT
stanovit
datum
dosud
žádný
návrat
stůl
vyjednávací
Žádné datum návratu k vyjednávacímu stolu nebylo dosud stanoveno .

Fig. 2. Tectogrammatical-layer alignment (only lemmas are depicted with the nodes)

type. In all other cases (at least one annotator makes any type of link), the connection is marked as *possible*.

We made no extra annotations for aligning tectogrammatical trees. Each node represents one content word on the surface. Thus we can transfer the manual word alignment to the alignment of tectogrammatical trees. Links from/to functional words (the tokens that do not have their own tectogrammatical nodes) are disregarded.

The transfer of alignment is illustrated in Figure 2. Each node has one reference to the original content word on the surface, which it got its lexical meaning from. Such references are depicted in Figure 2 by dashed lines.

Inter-annotator agreement (IAA) reflects the reliability of manual annotations. For evaluating IAA on the alignment task, we use the following formula:

$$IAA(A,B) = \frac{2 \cdot |L_A \cap L_B|}{|L_A| + |L_B|},$$

where L_A and L_B are sets of alignment connections made by annotator A and annotator B respectively. Since we have different types of connections, we can either distinguish the types in the evaluation (i.e., $L_A \cap L_B$ contains only connections that both annotators labeled equally), or disregard them. The third alternative is to regard only the sure

Table 1. Size of the manually aligned data

domain	sent.	words/sent.	
	pairs	CS .	EN
newspapers articles	984	21	24
E-books, short stories	500	18	20
EU-laws	501	21	27
PCEDT corpus	515	24	25

Table 2. Inter-annotator agreement

IAA [%]	all	content	funct.
types distinguished	83.3	90.9	76.8
types not dist.	89.6	94.6	84.2
sure links only	92.9	94.8	88.8

connections. The results of the three evaluation variants are summarized in Table 2. Connections between content and functional words are included within functional connections (the third column).

We can see that the alignment of the content words only is obviously less problematic for annotators compared to functional words. This evidence is in the same direction as that of Haruno and Yamazaki [5].

3 Evaluation Metrics

We randomly split the golden aligned data into two equal-sized parts, one for evaluating the performance of all aligners, and the other for training our tectogrammatical aligner.

For evaluating the quality of aligners we use alignment error rate (AER) described in [1], which combines precision and recall. Obviously, asserting connections that were neither sure nor possible causes lower precision, whereas omitting sure connections causes lower recall.

$$Prec = \frac{|(P \cup S) \cap A|}{|A|}, \quad Rec = \frac{|S \cap A|}{|S|}, \quad AER = 1 - \frac{|(P \cup S) \cap A| + |S \cap A|}{|S| + |A|}$$

where S is the set of *sure* links, P is the set of the *possible* links and A is the set of links suggested by the evaluated automatic aligner.

4 GIZA++ Word Alignment

We applied GIZA++[2] alignment tool [1] on data composed of two parts: 3,500,000 sentence pairs from the Czech-English parallel corpus CzEng [9], and 1,250 sentence pairs from the above mentioned evaluation data set. All texts were lemmatized[3] in both languages by lemmatizers available in TectoMT [10].

We ran GIZA++ in both directions (English to Czech and Czech to English) and symmetrized the outputs using intersection, grow-diag-final and union symmetrization, as described by Och and Ney [1]. The results are given in Table 3.

[2] Default settings, IBM models and iterations: $1^5 3^3 4^3$.

[3] Bojar and Prokopová [8] showed that lemmatization of the input text reduces the Czech vocabulary size to a half. Thus the vocabulary sizes of Czech and English become comparable. The data are thus not so sparse, which helps alignment error rate by about 10% absolute.

Table 3. GIZA++ evaluation table for all words (%)

symmetrization	precision	recall	AER
intersection	95.8	79.0	**13.2**
grow-diag-final	71.5	92.0	20.3
union	68.5	93.2	22.1

Table 4. GIZA++ evaluation table for content words only (%)

symmetrization	precision	recall	AER
intersection	97.8	82.2	**10.6**
grow-diag-final	78.5	93.6	14.7
union	74.3	94.7	17.1

We can see that while the *intersection* symmetrization is too sparse (precision is much higher than recall), the *grow-diag-final* and *union* symmetrizations have the opposite problem. We did not present *grow* and *grow-diag* symmetrizations – they are very similar to the *grow-diag-final* and have also much higher recall than precision. There is no symmetrization with the density of connections similar to our manually aligned data, the nearest one is the *intersection*.

We also measured how successful is GIZA++ on the content words only. We transferred the word alignment generated by GIZA++ to the tectogrammatical trees in the same way as we did it for the manual alignment (Figure 2). Table 4 shows that the best AER is achieved using the intersection symmetrization again.

5 Tectogrammatical Alignment

This section describes our approach to aligning content words using tectogrammatical tree structures.

At first we have to build the trees. All sentences in both languages are automatically parsed up to the t-layer using TectoMT[4] [10]. Czech sentences are first tokenized, morphologically analyzed and disambiguated by the morphological tagger shipped with Prague Dependency Treebank [4]. After that, the syntactic analysis realized by McDonald's MST parser [11] comes. The resulting analytical trees are then automatically converted (mostly by rule-based scripts) into tectogrammatical trees. English sentences are tokenized in the Penn Treebank style, tagged by the TnT tagger [12]. The analysis continues in the same way as for Czech – McDonald's MST parser and conversion to the tectogrammatical trees. Finally, after applying GIZA++ on the lemmatized surface sentences, the data are prepared for the tectogrammatical alignment.

```
foreach $(CT, ET) \in TreePairs$ do
    foreach $cnode \in CT$ do
        $counterpart(cnode) = \text{argmax}_{enode} \sum w_i^{c2e} \cdot f_i(cnode, enode)$;
    foreach $enode \in ET$ do
        $counterpart(enode) = \text{argmax}_{cnode} \sum w_i^{e2c} \cdot f_i(cnode, enode)$;
        if $counterpart(counterpart(enode)) = enode$ then Align($cnode$,$enode$);
```

Fig. 3. Pseudo-code for the first phase of t-alignment

[4] TectoMT is a software framework for developing machine translation systems.

Table 5. T-aligner evaluation (%)

alignment tool	prec.	recall	AER
GIZA++ (intersection)	97.8	82.2	10.6
T-aligner without GIZA	92.7	86.8	10.3
T-aligner using GIZA	96.0	89.7	**7.3**

Table 6. Improved GIZA++ word alignment evaluation (%)

alignment tool	prec.	recall	AER
GIZA++ (intersection)	95.8	79.0	13.2
improved GIZA++	94.3	84.6	**10.7**

Table 7. Features, their types, and weights in both directions

feature name	type	cs2en	en2cs
identical t-lemmas	binary	1.41	1.04
equal to 1 if Czech t-lemma is the same string as the English one			
verb, *adjective* **position similarity**	real	2.66	3.12
difference between relative linear positions of t-nodes			
t-lemma pair in dictionary	binary	1.88	2.06
equal to 1 if the pair of t-lemmas occurs in the translation dictionary			
3 letter match	binary	2.86	2.53
equal to 1 if the three-letter prefixes of Czech and English t-lemmas are identical			
equal number prefix	binary	9.58	7.00
Czech and English t-lemmas start with the same sequence of digits.			
aligned by GIZA++, direction en2cs	binary	1.20	2.31
equal to 1 if the corresponding surface words were aligned by GIZA++ (left)			
aligned by GIZA++, direction cs2en	binary	2.29	0.07
equal to 1 if the corresponding surface words were aligned by GIZA++ (right)			
aligned by GIZA++, intersection symmetrization	binary	1.57	0.92
equal to 1 if the corresponding surface words were aligned by GIZA++ (intersection)			
translation probability from dictionary	real	1.02	1.34
probability of Czech t-lemma, if the English t-lemma is given			
equal semantic part of speech	binary	1.61	1.06
equals to 1 if both semantic parts of speech (e.g. *noun*, *verb*, and *adjective*) are equal			
position similarity of parents nodes	real	0.76	0.92
difference between relative linear positions of parents			
parents aligned by GIZA++, intersection sym.	binary	0.21	0.28
equal to 1 if the surface words corresponding to parents were aligned by GIZA++			

The presented algorithm is based on a linear model and consists of three phases. First, we connect each English node with its most probable Czech counterpart. Second, we do the same for the opposite direction – we find the most probable English node for each Czech node. Finally, we make an intersection of these two alignments and declare it as our result. There is a pseudocode is in Figure 3. The most probable counterpart is the node with the highest score – the scalar product of vector of feature values and vector of feature weights. The weights for the Czech-to-English direction (w_i^{c2e}) are different from those for the opposite direction (w_i^{e2c}). We train the weights on the training part of manual aligned data and use an implementation of the discriminative reranker described by Collins in [13] (basically a modification of averaged perceptron).

Features are individual measurable properties of a pair of Czech and English node. We use features concerning similarities of lemmas and other t-node attributes, similarity in relative linear position of t-nodes within the sentences, and similarities of their child and parent nodes. There are several features taking into account whether GIZA++ aligned the examined pair on the surface or not; some features carry information from the probabilistic translation dictionary. This dictionary was compiled from parallel corpora PCEDT [7] and subsequently extended by word pairs acquired from the parallel corpus CzEng [9] aligned on the surface. All features are listed in Table 7. Their type and weights obtained from the reranker are also included.

The T-aligner evaluation and comparison with the results of GIZA++ (*intersection* symmetrization) in Table 5. Our T-aligner helps GIZA++ to decrease AER by 3.3% absolute. We also ran T-aligner without using GIZA++, AER reached 10.3 %, the error rate is therefore comparable to GIZA++. The advantage is that the combination of these two different approaches reaches lower AER.

6 Improving Surface Word Alignment

Our T-aligner has better results on the content words, when evaluated using our manual alignment. We can also measure how it can increase GIZA++ alignment on the surface. Since the GIZA++ word alignment has higher precision and lower recall (Table 3), we add all connections that were made by our T-aligner (it does not matter whether there was a connection made by GIZA++ or not) and no connections to delete. The results are in Table 6. The alignment error rate decreases by 2.5% absolute.

7 Conclusions

We described the algorithm for alignment of deep-syntactic dependency trees (tectogrammatical trees), where only content (autosemantic) words are aligned. We showed that alignment of content words is "simpler" both for the people (inter-annotator agreement is 5 % absolute higher) and for the automatic tools (GIZA++ AER decreases by 2.6 %). Our T-aligner outperformed GIZA++ in AER, however, the results are not straightforwardly comparable since GIZA++ could not be trained using the manually aligned data. We use aligned dependency trees for experimenting with statistical dependency-based MT. If we merge the best acquired alignment of content words with the GIZA++ surface word alignment, the resulting error-rate is lower by 2.5 %. It remains an open question how this "better" word alignment can improve phrase-based machine translation systems.

References

1. Och, F.J., Ney, H.: A Systematic Comparison of Various Statistical Alignment Models. Computational Linguistics 29(1), 19–51 (2003)
2. Menezes, A., Richardson, S.D.: A best-first alignment algorithm for automatic extraction of transfer mappings from bilingual corpora. In: Proceedings of the workshop on Data-driven methods in machine translation, vol. 14, pp. 1–8 (2001)
3. Sgall, P.: Generativní popis jazyka a česká deklinace. Academia, Prague (1967)

4. Hajič, J., Hajičová, E., Panevová, J., Sgall, P., Pajas, P., Štěpánek, J., Havelka, J., Mikulová, M.: Prague Dependency Treebank 2.0. Linguistic Data Consortium, LDC Catalog No.: LDC2006T01, Philadelphia (2006)
5. Haruno, M., Yamazaki, T.: High-performance Bilingual Text Alignment Using Statistical and Dictionary Information. In: Proceedings of the 34th conference of the Association for Computational Linguistics, pp. 131–138 (1996)
6. Watanabe, H., Kurohashi, S., Aramaki, E.: In: Finding Translation Patterns from Paired Source and Target Dependency Structures, pp. 397–420. Kluwer Academic, Dordrecht (2003)
7. Cuřín, J., Čmejrek, M., Havelka, J., Hajič, J., Kuboň, V., Žabokrtský, Z.: Prague Czech-English Dependency Treebank, Version 1.0. Linguistics Data Consortium, Catalog No.: LDC2004T25 (2004)
8. Bojar, O., Prokopová, M.: Czech-English Word Alignment. In: Proceedings of the Fifth International Conference on Language Resources and Evaluation (LREC 2006), ELRA, May 2006, pp. 1236–1239 (2006)
9. Bojar, O., Janíček, M., Žabokrtský, Z., Češka, P., Beňa, P.: CzEng 0.7: Parallel Corpus with Community-Supplied Translations. In: Proceedings of the Sixth International Language Resources and Evaluation (LREC'08), Marrakech, Morocco, ELRA (May 2008)
10. Žabokrtský, Z., Ptáček, J., Pajas, P.: TectoMT: Highly Modular MT System with Tectogrammatics Used as Transfer Layer. In: Proceedings of the 3rd Workshop on Statistical Machine Translation, ACL (2008)
11. McDonald, R., Pereira, F., Ribarov, K., Hajič, J.: Non-Projective Dependency Parsing using Spanning Tree Algorithms. In: Proceedings of Human Langauge Technology Conference and Conference on Empirical Methods in Natural Language Processing (HTL/EMNLP), Vancouver, BC, Canada, pp. 523–530 (2005)
12. Brants, T.: TnT - A Statistical Part-of-Speech Tagger. In: Proceedings of the 6th Applied Natural Language Processing Conference, Seattle, pp. 224–231 (2000)
13. Collins, M.: Discriminative Training Methods for Hidden Markov Models: Theory and Experiments with Perceptron Algorithms. In: Proceedings of EMNLP, vol. 10, pp. 1–8 (2002)